



**Università degli Studi di Salerno  
Anno Accademico 2018/2019**

**Corso di Ingegneria del Software**

**Problem Statement  
v 1.2**



## Sommario

INTRODUZIONE.....	3
Descrizione del sistema.....	4
Scopo del sistema .....	4
Obiettivi.....	4
Overview .....	4
Architettura del sistema corrente .....	5
Sistema proposto .....	5
Identificazione attori.....	5
Requisiti funzionali.....	5
Requisiti non funzionali.....	9
Modelli del sistema .....	10
Scenari amministratore.....	10
Sequence diagram SC_2.1.....	10
Sequence diagram SC_2.3.....	12
Sequence diagram SC_2.4.....	13
Scenari cliente .....	14
Sequence diagram SC_1.1.....	14
Sequence diagram SC_1.2.....	15
Sequence diagram SC_1.3.....	16
Sequence diagram SC_1.4.....	17
Sequence diagram SC_1.5.....	18
Casi d'uso.....	19
Schema generale dei casi d'uso .....	19
Use case utente generico: amministratore, cliente.....	20
Login .....	20

## Top Manager:

*Prof. De Lucia Andrea*

## Team di sviluppo:

Nome e Cognome	Matricola
<i>Aniello Mancusi</i>	0512102610
<i>Vincenzo Zito</i>	0512100507

## Revision History:

Autore	Data	Descrizione	Versione
<i>Aniello Mancusi</i>	<i>23/02/2019</i>	<i>Struttura documento</i>	<i>v 1.0</i>
<i>Vincenzo Zito</i>	<i>24/02/2019</i>	<i>Stesura generale del documento</i>	<i>v 1.1</i>
<i>Aniello Mancusi</i>	<i>26/02/2019</i>	<i>Modelli del sistema, Use case diagram, Mockup utente generico, Inserimento Object boundary, Object identity, Control Object</i>	<i>v 1.2</i>

# INTRODUZIONE

## Descrizione del sistema

Sistema informatizzato di gestione di un negozio di informatica. Si prevede la realizzazione di un database che permetta la catalogazione digitale dei dipendenti, dei prodotti e servizi offerti e dei clienti. In secondo luogo, poi, la realizzazione di una piattaforma web che permetta la gestione del negozio da parte di un amministratore e la possibilità di acquisto di beni e servizi da parte dei clienti.

## Scopo del sistema

Il sistema si propone come obiettivo quello di rendere il carico di lavoro dell'amministratore più fluido e leggero, automatizzando la maggior parte delle operazioni. Le operazioni di carico merci, di acquisto e vendita prodotti saranno gestite dal sistema, i dati verranno conservati in un database relazionale creato ad-hoc e la gestione sarà affidata ad una web application disegnata su misura per il cliente.

## Obiettivi

L'obiettivo che ci poniamo è quello di fornire una piattaforma semplice ed intuitiva sia per chi è già cliente sia per chi gestisce lo stesso.

Le macro funzionalità che vogliamo offrire sono:

- Cliente: registrazione, login e logout, acquisto, gestione ordine;
- Amministratore: login e logout, carico merci, controllo clienti, controllo acquisti;

## Overview

Il presente documento R.A.D. (Requirements Analysis Document) descrive il sistema in termini di requisiti funzionali e serve come base contrattuale con il cliente.

## Architettura del sistema corrente

L'architettura attualmente proposta non va a sostituire nessuna architettura presente perché il sistema è alla sua prima versione.

## Sistema proposto

### Identificazione attori

Amministratore: gestisce la piattaforma, inizialmente si occuperà di ogni aspetto funzionale del sistema ma è previsto, in un secondo momento, lo sviluppo di un modulo che possa prevedere la divisione dei compiti tra i vari attori che gestiscono separatamente i vari aspetti che sono presenti (es. vendita, gestione del magazzino, spedizioni).

### Requisiti funzionali

In questa sezione vengono specificati quali funzioni il sistema deve fornire per soddisfare i bisogni dei clienti e del gestore del negozio.

#### **RF\_1- Gestione Registrazione**

Questa funzionalità consente ai clienti di registrarsi ad StaySoftware inserendo i dati richiesti.

**Attore:** Cliente

##### ***RF\_1.1 – Registrazione cliente:***

Questa funzionalità permette ad un cliente di registrarsi al sistema in modo da poter effettuare il login per poi eseguire l'acquisto di beni e servizi.

#### **RF\_2- Gestione Autenticazione**

Questa funzionalità è comune per tutti gli attori e deve essere in grado di gestire l'autenticazione al sistema degli stessi.

##### ***RF\_2.1 – Login:***

Questa funzionalità permette di far effettuare l'accesso al sistema autenticandosi ed avendo a disposizione le varie funzionalità offerte dalla piattaforma.

##### ***RF 2.2 – Logout:***

Questa funzionalità permette la disconnessione dal sistema.

## **RF\_3 – Gestione utenti**

Questa funzionalità permette la gestione degli utenti registrati alla piattaforma.

**Attore:** Amministratore

### **RF\_3.1.1 – Modifica profilo:**

Questa funzionalità permette all'amministratore la modifica di alcuni parametri del profilo.

### **RF\_3.1.2 – Visualizza profilo:**

Questa funzionalità permette all'amministratore la visualizzazione del proprio profilo.

### **RF\_3.1.3 - Visualizza profilo clienti:**

Questa funzionalità permette all'amministratore la visualizzazione del profilo dei clienti registrati alla piattaforma.

**Attore:** Cliente

### **RF\_3.2.1 - Modifica profilo:**

Questa funzionalità permette al cliente la modifica di alcuni parametri del profilo.

### **RF\_3.2.2 - Visualizza profilo:**

Questa funzionalità permette al cliente la visualizzazione del profilo.

## **RF\_4 - Gestione Ordine**

Questa funzionalità permette la gestione degli acquisti.

**Attore:** Cliente

### **RF\_4.1 - Visualizza prodotti:**

Questa funzionalità permette al cliente di navigare nel sito visualizzando i prodotti in vendita.

### **RF\_4.2 - Carrello:**

Questa funzionalità permette al cliente di navigare nel sito inserendo nel carrello i prodotti che vuole acquistare.

### **RF\_4.3 - Acquisto:**

Questa funzionalità permette al cliente di finalizzare l'acquisto dei prodotti.

#### **RF\_4.4 - Visualizza cronologia acquisti:**

Questa funzionalità permette al cliente la visualizzazione della cronologia dei suoi acquisti.

#### **RF\_5 - Gestione Prodotti**

Questa funzionalità permette la gestione del magazzino.

**Attore:** Amministratore

##### **RF\_5.1 - Inserimento prodotto:**

Questa funzionalità permette all'amministratore di aggiornare la lista dei prodotti presenti nel negozio.

##### **RF\_5.2 - Modifica quantità prodotti:**

Questa funzionalità permette all'amministratore di modificare la quantità dei prodotti.

##### **RF\_5.2 - Modifica prezzo prodotti:**

Questa funzionalità permette all'amministratore di modificare il prezzo dei prodotti.

#### **RF\_6 - Gestione Riparazioni**

Questa funzionalità permette la gestione delle riparazioni.

**Attore:** Amministratore

##### **RF\_6.1.1 - Invio preventivo:**

Questa funzionalità permette all'amministratore di inoltrare un preventivo rispetto ad una richiesta di riparazione che ha ricevuto.

##### **RF\_6.1.2 - Inizio riparazione:**

Questa funzionalità permette all'amministratore di avvertire il cliente che la riparazione è in corso.

##### **RF\_6.1.3 - Riparazione effettuata:**

Questa funzionalità permette all'amministratore di avvertire il cliente che la riparazione è stata effettuata con successo.

**Attore:** Cliente

**RF\_6.2.1 - Richiesta preventivo:**

Questa funzionalità permette all'utente di inoltrare una richiesta per un preventivo.

**RF\_6.2.2 - Accettazione preventivo:**

Questa funzionalità permette all'utente di accettare un preventivo che gli è stato inoltrato.

**RF\_6.2.3 - Rifiuto preventivo:**

Questa funzionalità permette all'utente di rifiutare un preventivo che gli è stato inoltrato.



## Requisiti non funzionali

### RNF\_1 - Usabilità

Il sistema deve rispettare il principio di usabilità che può essere articolato come segue:

- **Sistema Efficace:** il sistema deve fornire strumenti precisi e completi con cui l'utente può raggiungere velocemente un obiettivo specifico.
- **Sistema Efficiente:** attraverso l'utilizzo di risorse competitive e complete che permettono agli utenti maggiore velocità nell'utilizzo della piattaforma.
- **Sistema Soddisfacente:** il sistema deve rilasciare feedback continui che permettono all'utente di orientarsi al meglio e di poter avere meno problemi possibili durante l'utilizzo della piattaforma.

### RNF\_2 - Affidabilità

Il sistema è basato sull'architettura Client-Server, entrambi i componenti devono essere affidabili quindi devono essere in grado di poter mantenere i propri dati anche in caso di guasti (problemi di fornitura elettrica, usura dell'hardware del server, attacchi informatici, Problemi legati al browser, interruzione della connettività alla rete ecc.).

Il sistema dev'essere inoltre usufruibile 24 ore su 24.

Il sistema deve prevedere periodicamente la possibilità di effettuare dei backup.

### RNF\_3 - Performance

Il sistema, inizialmente, deve essere in grado di gestire 1000 connessioni contemporanee.

Deve quindi poter gestire insieme vendite e caricamento merce senza che gli utilizzatori del sistema si accorgano del carico di lavoro che il server gestisce.

### RNF\_4 - Manutenibilità

Il sistema verrà implementato con un'architettura three-tier, così eventuali modifiche ad uno dei tre livelli non comporterà il blocco del sistema e la relativa modifica dei blocchi rimanenti rispetto a quello aggiornato.

### RNF\_5 - Implementazione

Back-end del sistema realizzato in Php.

Base di dati realizzata con il database relazionale MySQL.

Front-end realizzato con l'utilizzo di un software per lo sviluppo web sfruttando i linguaggi HTML, CSS.

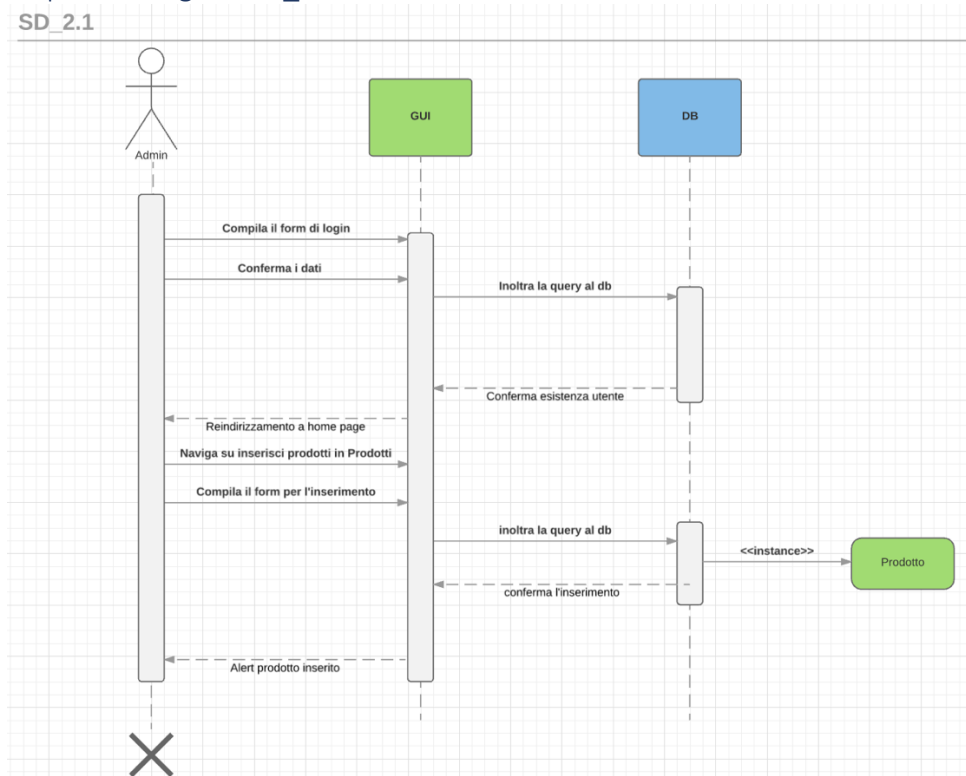
## Modelli del sistema

Si prevedono inizialmente due tipi di attori: amministratore e cliente.

### Scenari amministratore

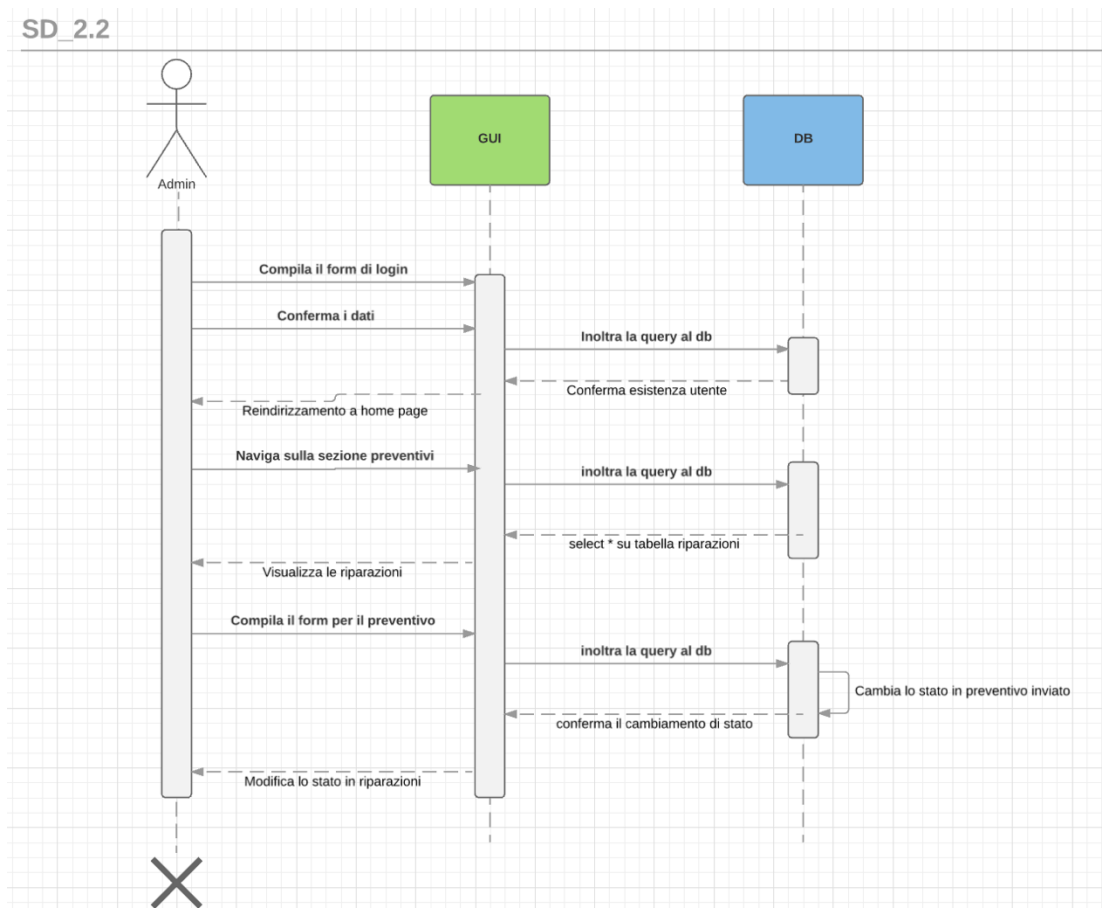
<b>Nome scenario</b>	SC_2.1_Inserimento_prodotti
<b>Partecipanti</b>	Marco: Amministratore
<b>Flusso di eventi</b>	<ol style="list-style-type: none"><li>1. Marco riceve la merce dal fornitore;</li><li>2. Marco controlla la merce;</li><li>3. Marco decide quindi di accedere alla piattaforma e destinare un certo quantitativo di prodotti alla vendita online;</li><li>4. Marco si collega a <a href="http://www.staysoftware.it.it">www.staysoftware.it.it</a>;</li><li>5. Marco compila il form di accesso inserendo username e password;</li><li>6. Marco conferma il login;</li><li>7. Il sistema verifica i dati e se validi permette l'accesso a Marco;</li><li>8. Marco seleziona dal menù prodotti la voce "inserisci prodotto";</li><li>9. Marco inizia a inserire tutti i prodotti ricevuti dal fornitore nel database compilando il form;</li><li>10. Completato l'inserimento Marco esegue il logout.</li><li>11. Marco decide quindi di controllare se ha inserito tutti i prodotti correttamente e si reca sulla pagina prodotti controllandone il contenuto.</li></ol>

### Sequence diagram SC\_2.1



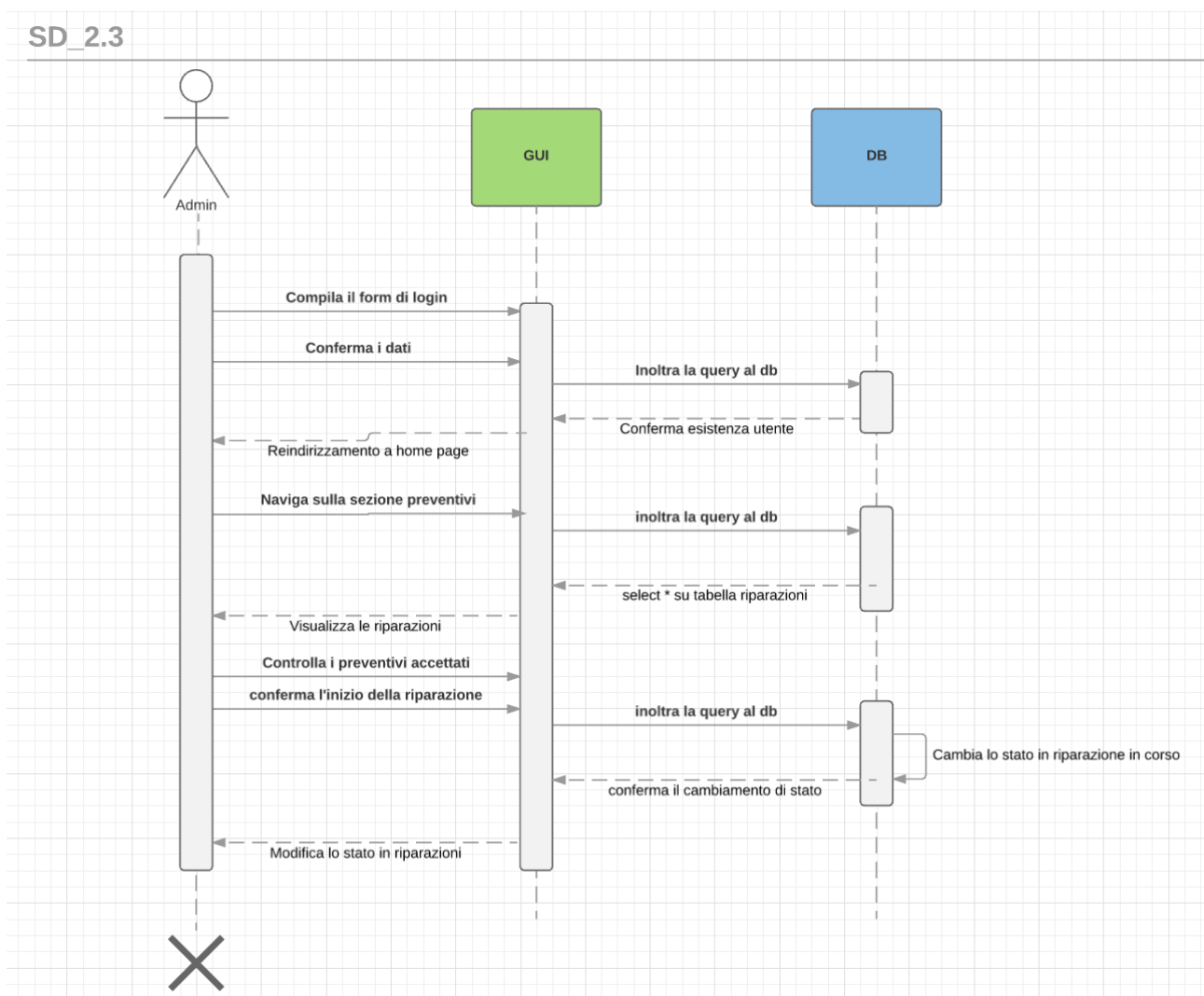
<b>Nome scenario</b>	SC_2.2_Invio_preventivo
<b>Partecipanti</b>	Marco: Amministratore
<b>Flusso di eventi</b>	<ol style="list-style-type: none"> <li>1. Marco decide di accedere alla piattaforma per controllare se ci sono richieste di preventivo;</li> <li>2. Marco si collega a <a href="http://www.staysoftware.it">www.staysoftware.it</a>;</li> <li>3. Marco compila il form di accesso inserendo username e password;</li> <li>4. Marco conferma il login;</li> <li>5. Il sistema verifica i dati e se validi permette l'accesso a Marco;</li> <li>6. Marco seleziona dal menù riparazioni la voce "richieste di riparazione";</li> <li>7. Legge le richieste che ci sono in sospeso e invia ai vari clienti il preventivo che ha stilato per la riparazione;</li> <li>8. Marco decide di disconnettersi.</li> </ol>

Sequence diagram SC\_2.2



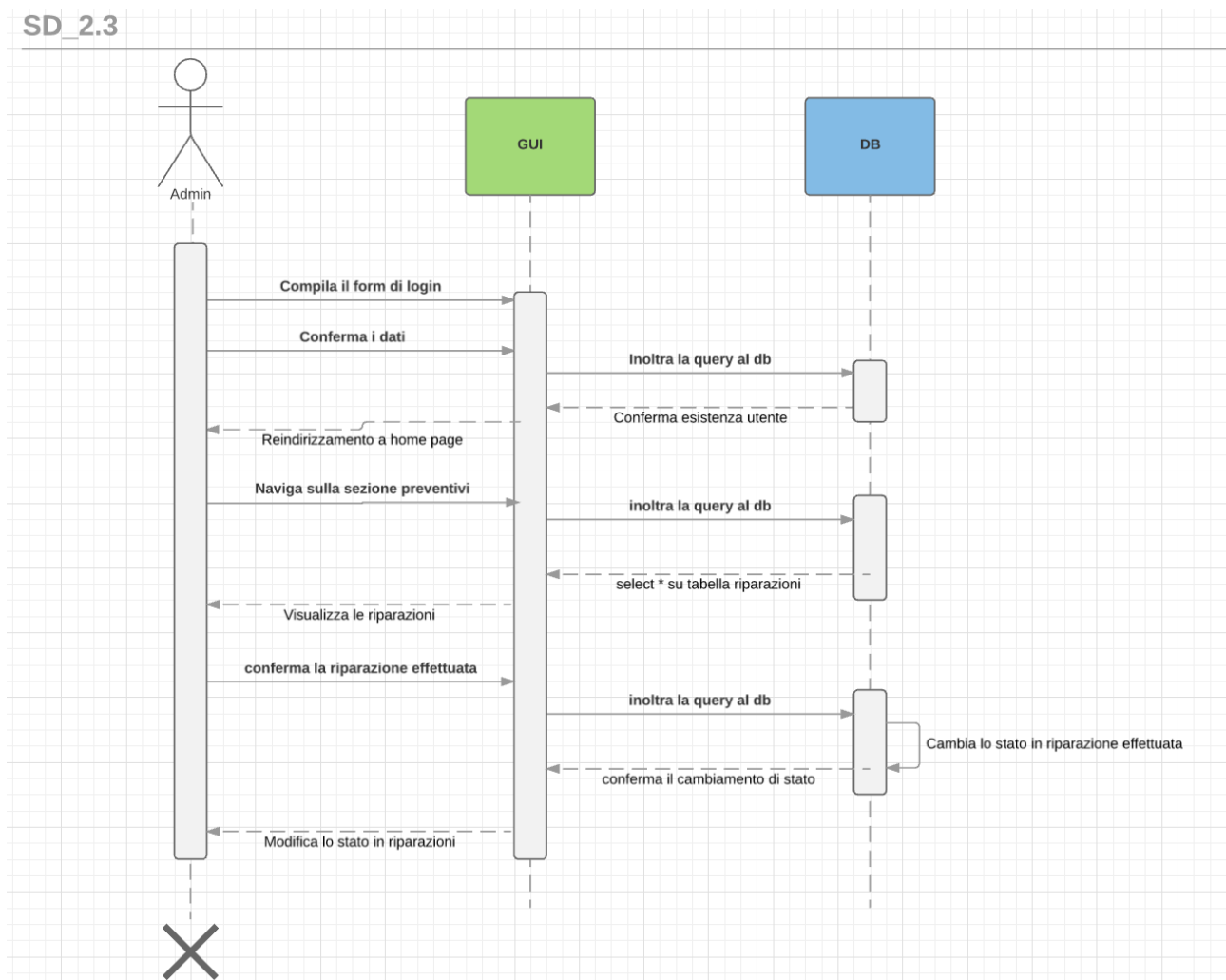
<b>Nome scenario</b>	SC_2.3_Riparazione_prodotto
<b>Partecipanti</b>	Marco: Amministratore
<b>Flusso di eventi</b>	<ol style="list-style-type: none"> <li>1. Marco decide di accedere alla piattaforma per controllare se i preventivi inviati sono stati accettati o rifiutati;</li> <li>2. Marco si collega a <a href="http://www.staysoftware.it">www.staysoftware.it</a>;</li> <li>3. Marco compila il form di accesso inserendo username e password;</li> <li>4. Marco conferma il login;</li> <li>5. Il sistema verifica i dati e se validi permette l'accesso a Marco;</li> <li>6. Marco seleziona dal menù riparazioni la voce "richieste di riparazione";</li> <li>7. Marco controlla i preventivi inviati;</li> <li>8. Nel caso in cui ci sono preventivi accettati Marco aspetta che i prodotti arrivino in negozio;</li> <li>9. Una volta ricevuti Marco invia un messaggio al cliente in cui specifica l'avvenuto inizio della riparazione;</li> <li>10. Marco decide di disconnettersi.</li> </ol>

Sequence diagram SC\_2.3



<b>Nome scenario</b>	SC_2.4_Riparazione_prodotto_effettuata
<b>Partecipanti</b>	Marco: Amministratore
<b>Flusso di eventi</b>	<ol style="list-style-type: none"> <li>1. Marco decide di accedere alla piattaforma per comunicare al cliente che la riparazione è stata effettuata;</li> <li>2. Marco si collega a <a href="http://www.staysoftware.it">www.staysoftware.it</a>;</li> <li>3. Marco compila il form di accesso inserendo username e password;</li> <li>4. Marco conferma il login;</li> <li>5. Il sistema verifica i dati e se validi permette l'accesso a Marco;</li> <li>6. Marco seleziona dal menù riparazioni la voce "richieste di riparazione";</li> <li>7. Marco scorre la pagina e trova la riparazione effettuata;</li> <li>8. Marco preme sul pulsante "Conferma riparazione in corso";</li> <li>9. Marco decide di disconnettersi.</li> </ol>

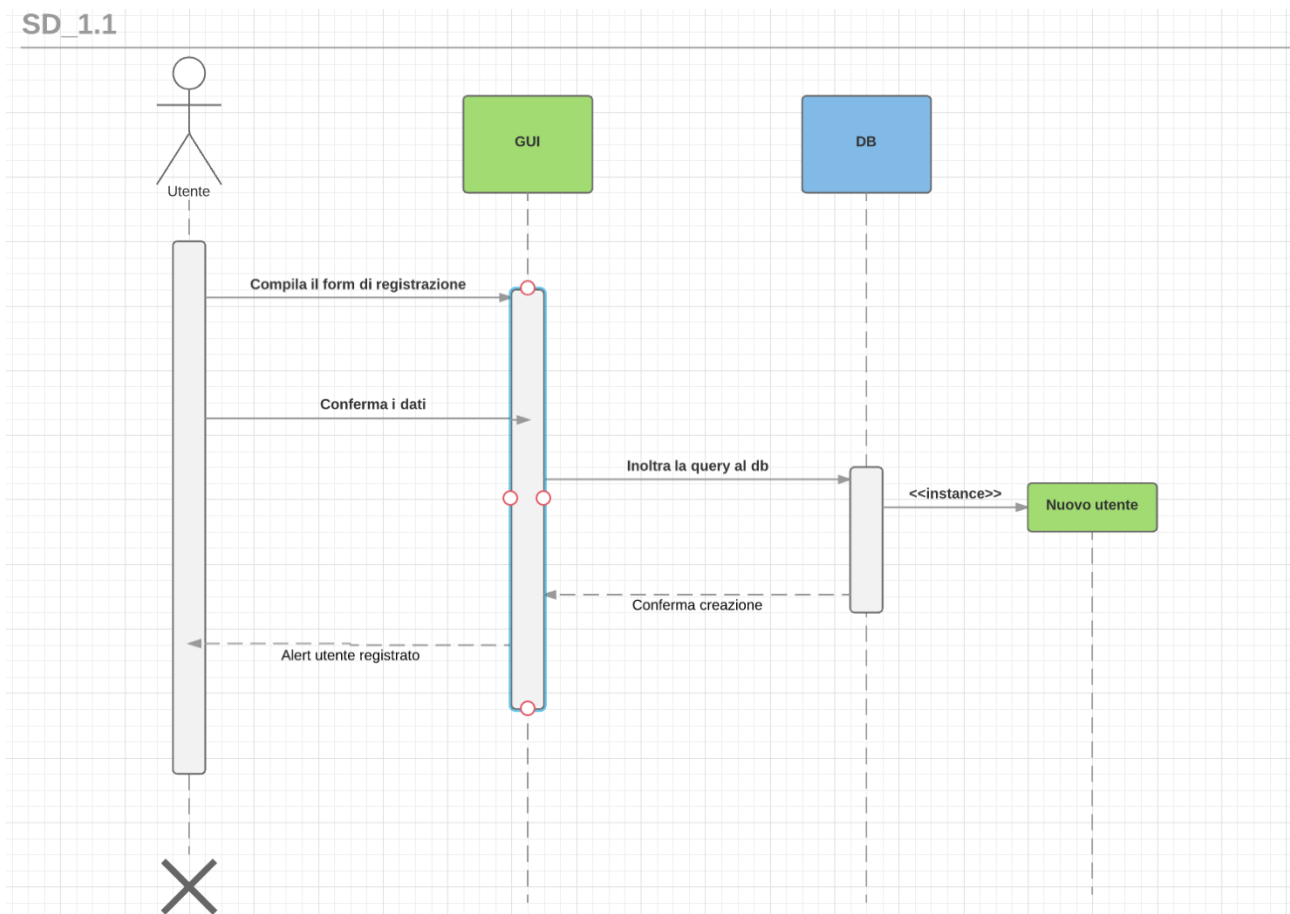
Sequence diagram SC\_2.4



## Scenari cliente

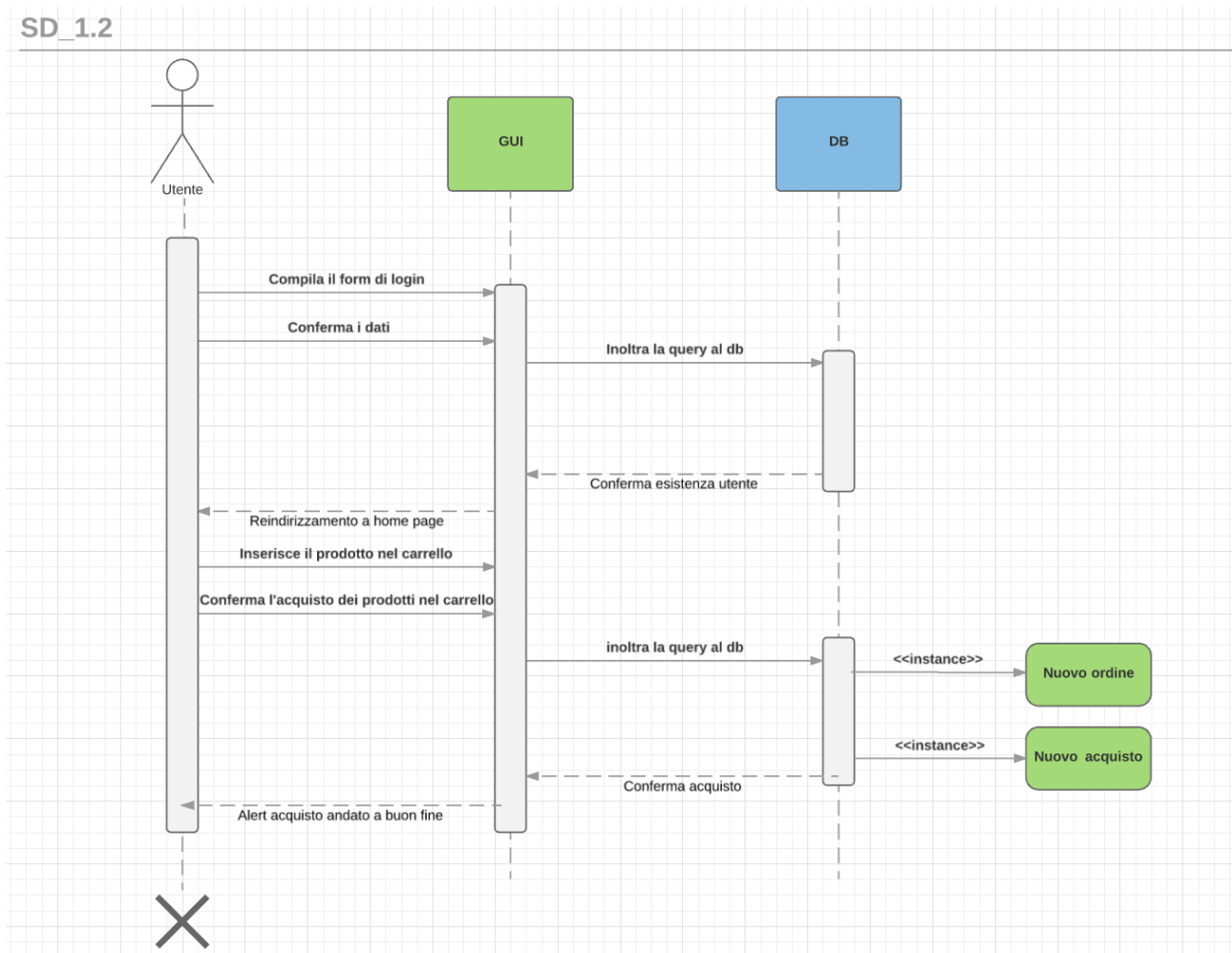
<b>Nome scenario</b>	SC_1.1_Registrazione_utente
<b>Partecipanti</b>	Vittorio: cliente
<b>Flusso di eventi</b>	<ol style="list-style-type: none"> <li>1. Vittorio accede per la prima volta a <a href="http://www.staysoftware.it">www.staysoftware.it</a>;</li> <li>2. Naviga nel sito ed entra nella sezione dedicata ai computer portatili;</li> <li>3. Scorre tra i vari modelli e trova il computer portatile adatto alle sue esigenze e decide di procedere con l'acquisto;</li> <li>4. Non essendo registrato al sito non può inserire l'oggetto nel carrello e quindi acquistare;</li> <li>5. Vittorio decide quindi di registrarsi dato che non si è mai registrato alla piattaforma;</li> <li>6. Vittorio compila il form di registrazione inserendo tutti i dati obbligatori;</li> <li>7. Vittorio conferma la registrazione.</li> </ol>

Sequence diagram SC\_1.1



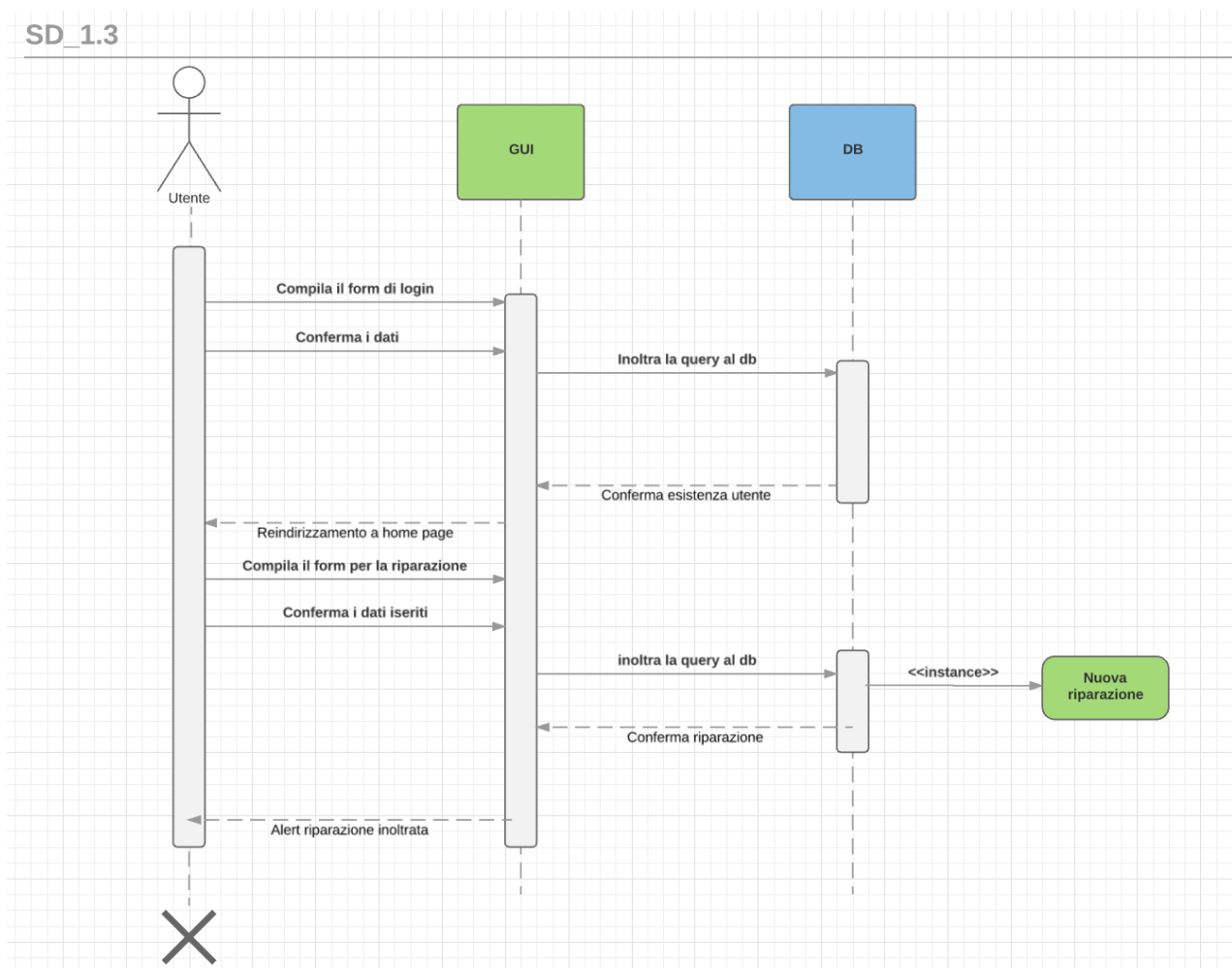
<b>Nome scenario</b>	SC_1.2_Acquisto
<b>Partecipanti</b>	Vittorio: cliente
<b>Flusso di eventi</b>	<ol style="list-style-type: none"> <li>1. Vittorio si collega a <a href="http://www.staysoftware.it">www.staysoftware.it</a>;</li> <li>2. Vittorio compila il form di accesso inserendo username e password;</li> <li>3. Vittorio conferma l'accesso;</li> <li>4. Vittorio naviga nella sezione prodotti dedicata ai computer;</li> <li>5. Dopo un'accurata ricerca trova il portatile adatto alle sue esigenze;</li> <li>6. Inserisce 1 come quantità e preme aggiungi a carrello;</li> <li>7. Vittorio decide di voler acquistare un solo pc e quindi si reca nella pagina carrello per poter procedere;</li> <li>8. Vittorio procede all'acquisto;</li> <li>9. Il sistema aggiorna il database.</li> </ol>

Sequence diagram SC\_1.2



<b>Nome scenario</b>	SC_1.3_Richiesta_preventivo_riparazione
<b>Partecipanti</b>	Vittorio: cliente
<b>Flusso di eventi</b>	<ol style="list-style-type: none"> <li>1. Vittorio vuole che gli venga riparato uno smartphone;</li> <li>2. Vittorio si collega a <a href="http://www.staysoftware.it">www.staysoftware.it</a>;</li> <li>3. Vittorio compila il form di accesso inserendo username e password;</li> <li>4. Vittorio conferma l'accesso;</li> <li>5. Vittorio accede ad I-Tech;</li> <li>6. Vittorio sceglie la voce "richiesta riparazione" dal menù riparazioni;</li> <li>7. Vittorio compila il form per la riparazione inserendo la descrizione del problema;</li> <li>8. Vittorio inoltra la domanda di riparazione;</li> <li>9. Il sistema informa l'utente che la richiesta è stata presa in carico e che il preventivo di riparazione verrà inviato all'utente entro 24h lavorative;</li> <li>10. Vittorio si disconnette</li> </ol>

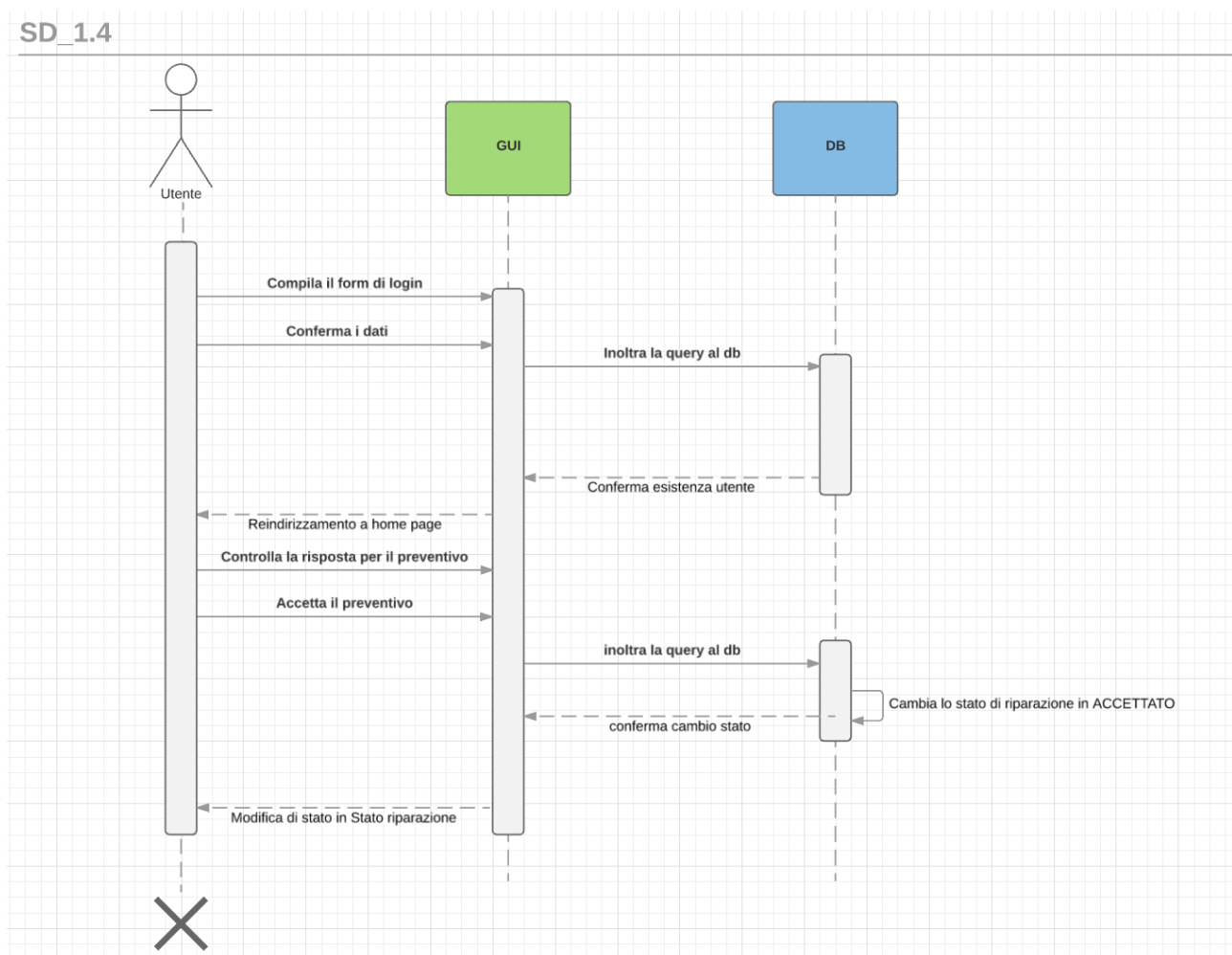
Sequence diagram SC\_1.3





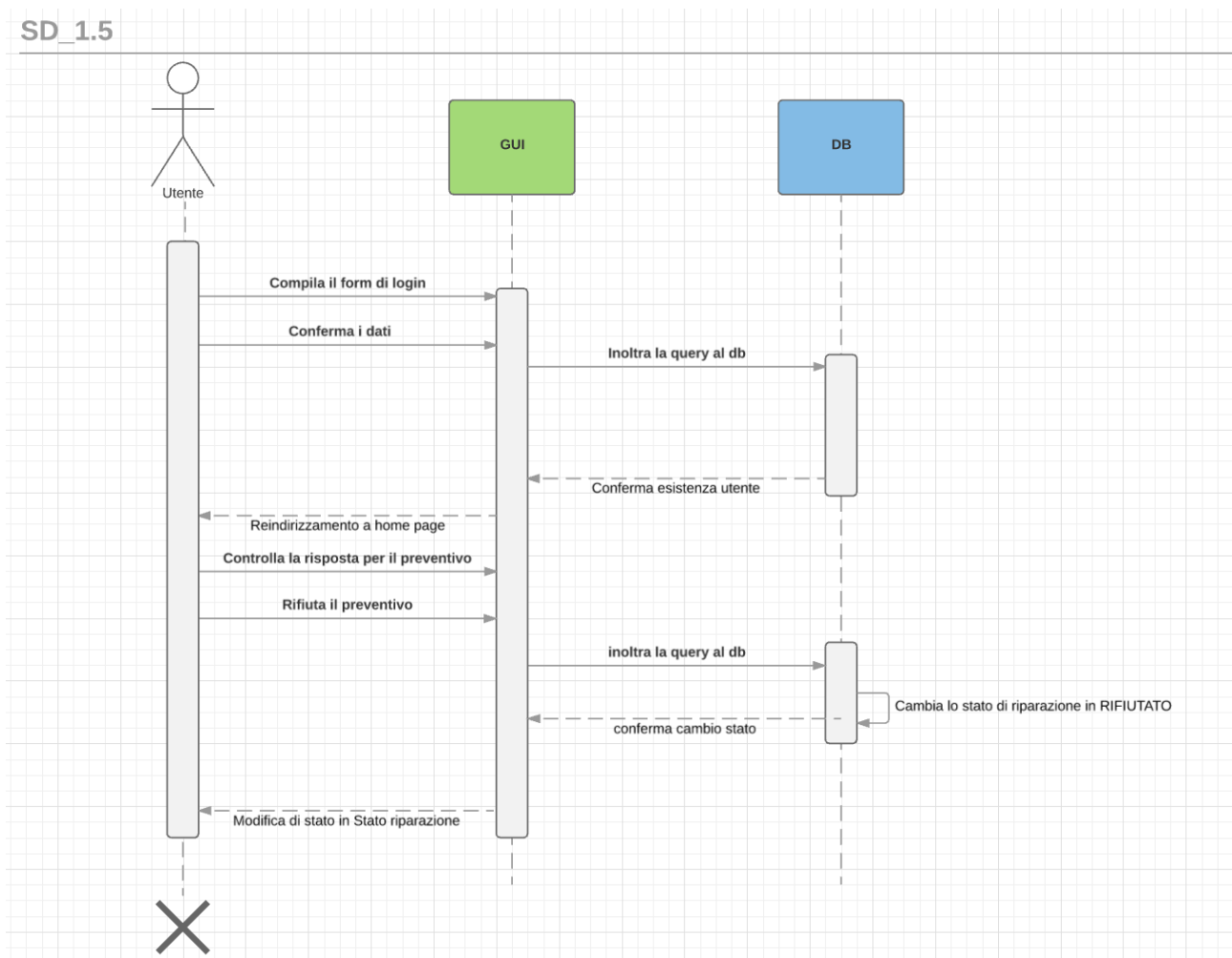
<b>Nome scenario</b>	SC_1.4_Accettazione_preventivo_riparazione
<b>Partecipanti</b>	Vittorio: cliente
<b>Flusso di eventi</b>	<ol style="list-style-type: none"> <li>1. Vittorio vuole controllare lo stato della richiesta preventivo effettuata;</li> <li>2. Vittorio si collega a <a href="http://www.staysoftware.it">www.staysoftware.it</a>;</li> <li>3. Vittorio compila il form di accesso inserendo username e password;</li> <li>4. Vittorio conferma l'accesso;</li> <li>5. Vittorio accede ad I-Tech;</li> <li>6. Vittorio sceglie la voce "le mie riparazioni" dal menù riparazioni;</li> <li>7. Nella sezione "prezzo" della riparazione viene visualizzato il prezzo della stessa;</li> <li>8. Vittorio controlla il preventivo e decide di riparare il telefono;</li> <li>9. Vittorio accetta quindi il preventivo, seleziona "Accetta" e ok.</li> <li>10. Il sistema conferma l'accettazione della riparazione;</li> <li>11. Vittorio si disconnette.</li> </ol>

Sequence diagram SC\_1.4



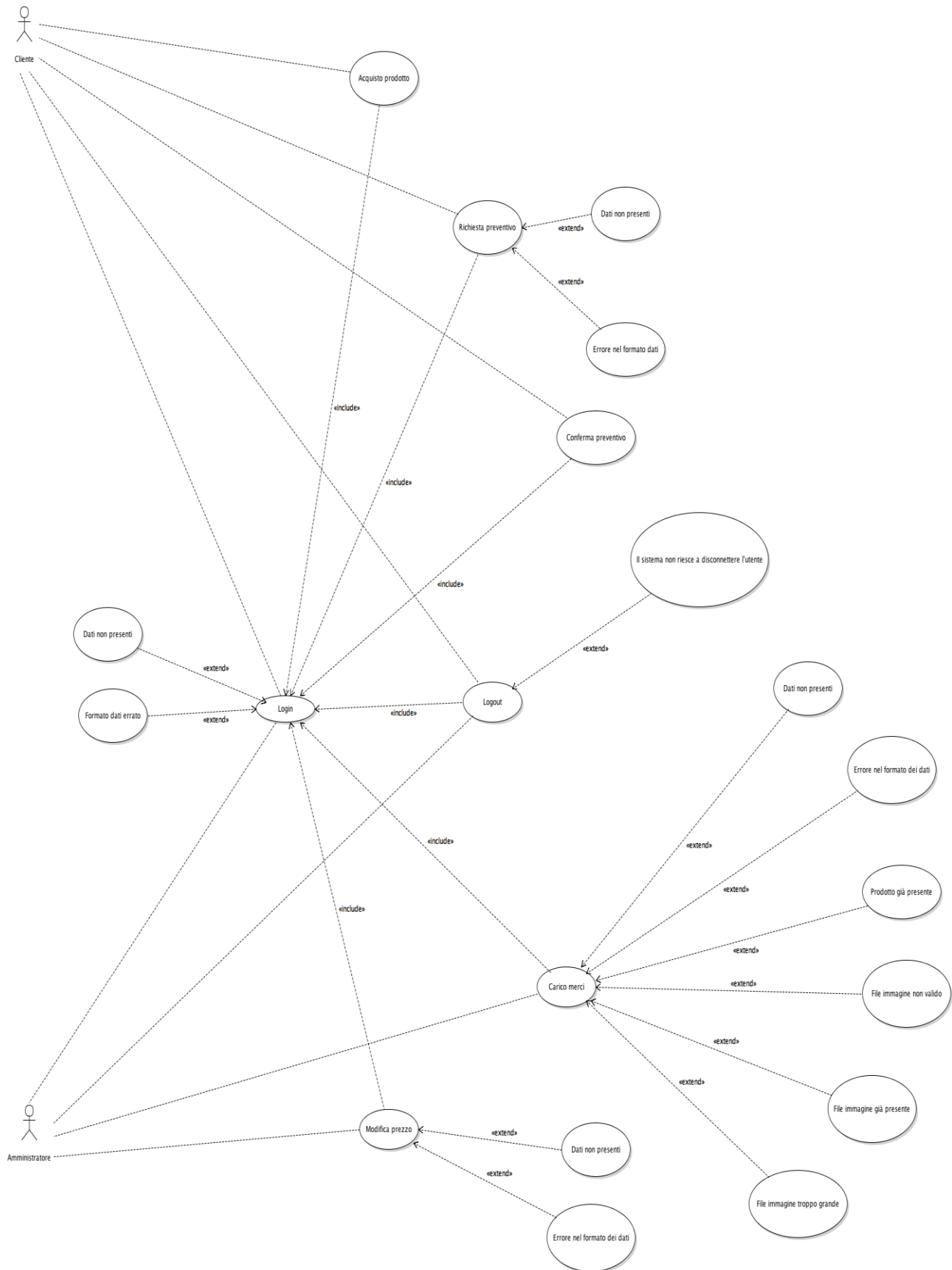
<b>Nome scenario</b>	SC_1.5_Rifiuto_preventivo_riparazione
<b>Partecipanti</b>	Vittorio: cliente
<b>Flusso di eventi</b>	<ol style="list-style-type: none"> <li>1. Vittorio vuole controllare lo stato della richiesta preventivo effettuata;</li> <li>2. Vittorio si collega a <a href="http://www.staysoftware.it">www.staysoftware.it</a>;</li> <li>3. Vittorio compila il form di accesso inserendo username e password;</li> <li>4. Vittorio conferma l'accesso;</li> <li>5. Vittorio accede ad I-Tech;</li> <li>6. Vittorio sceglie la voce "le mie riparazioni" dal menù riparazioni;</li> <li>7. Nella sezione "prezzo" della riparazione viene visualizzato il prezzo della stessa;</li> <li>8. Vittorio controlla il preventivo e decide non di riparare il telefono;</li> <li>9. Vittorio accetta quindi il preventivo, seleziona "Rifiuta" e ok.</li> <li>10. Il sistema conferma il rifiuto della riparazione;</li> <li>11. Vittorio si disconnette.</li> </ol>

Sequence diagram SC\_1.5



## Casi d'uso

### Schema generale dei casi d'uso



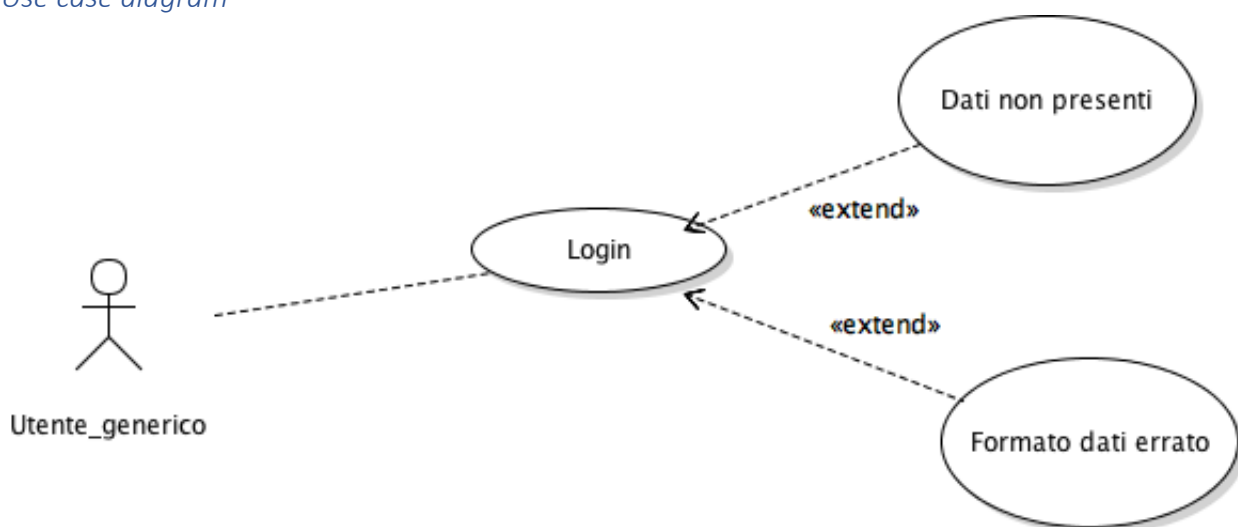
## Use case utente generico: amministratore, cliente

### Login

#### Use case

<b>Nome use case</b>	Login
<b>Partecipanti</b>	Utente generico
<b>Condizioni di ingresso</b>	L'utente accede al sistema L'utente è registrato al sistema
<b>Flusso di eventi</b>	<ol style="list-style-type: none"><li>1. L'utente accede al sistema tramite motore di ricerca o tramite link diretto;</li><li>2. Il sistema mostra un form contenente due campi: "Username" e "Password", inoltre mostra due pulsanti: "Login" e "Registrati".</li><li>3. L'utente generico inserisce username e password e sottomette il form cliccando su "Login";</li><li>4. Il sistema riceve i dati e cerca l'username e la password inserita nel database;</li><li>5. Il sistema elabora i dati e mostra la pagina iniziale del sito.</li></ol>
<b>Eccezioni</b>	Nel punto 2 potrebbero non esserci dati presenti Nel punto 2 potrebbero esserci dati in formato errato Nel punto 4 il sistema potrebbe non trovare corrispondenza nel database
<b>Condizioni di uscita</b>	L'utente ha avuto accesso al sistema

#### Use case diagram

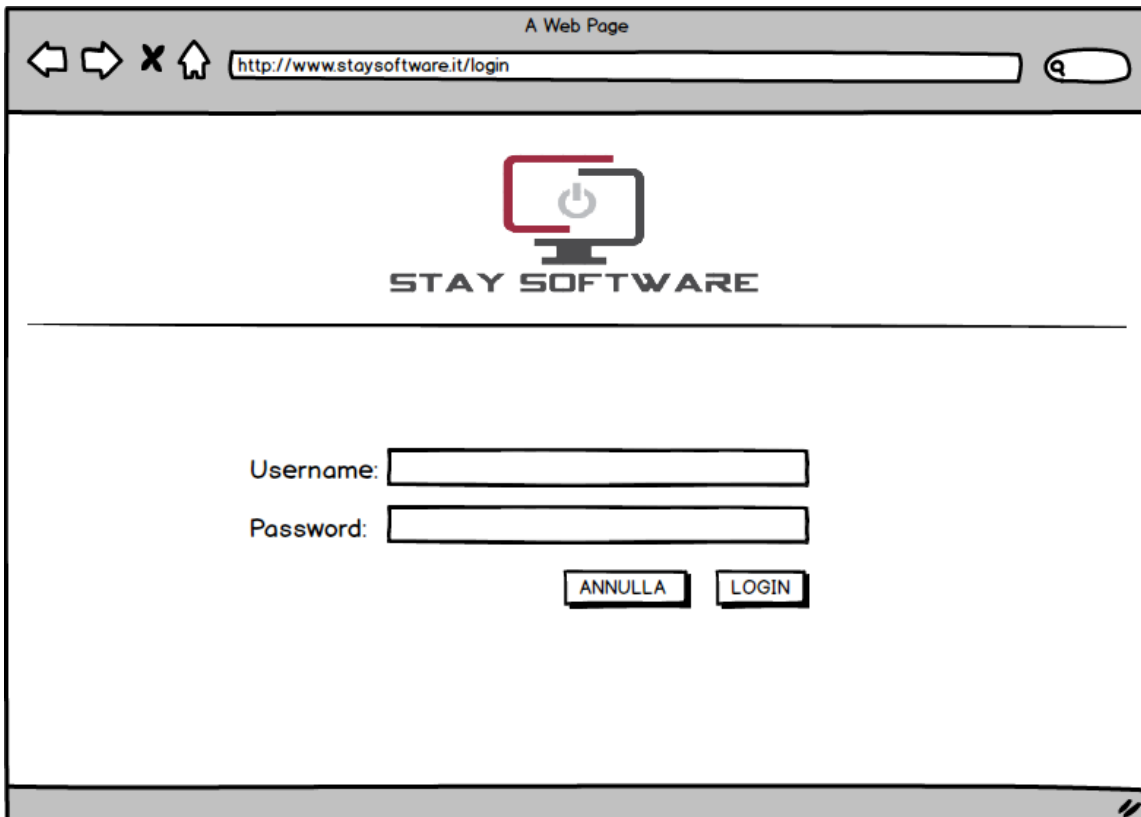


Mockup – cliente

1.



2.



3.

A Web Page

http://www.staysoftware.it/login

**STAY SOFTWARE**

Username:

Password:

3.1 Errore

A Web Page

http://www.staysoftware.it/login\_error

**STAY SOFTWARE**

Username:

Password:

**Attenzione**  
Username o password errati

4.



1.

A Web Page

http://www.staysoftware.it/admin



STAY SOFTWARE

---

## Pannello di Controllo

Username:

Password:

2.

A Web Page

http://www.staysoftware.it/admin



STAY SOFTWARE

---

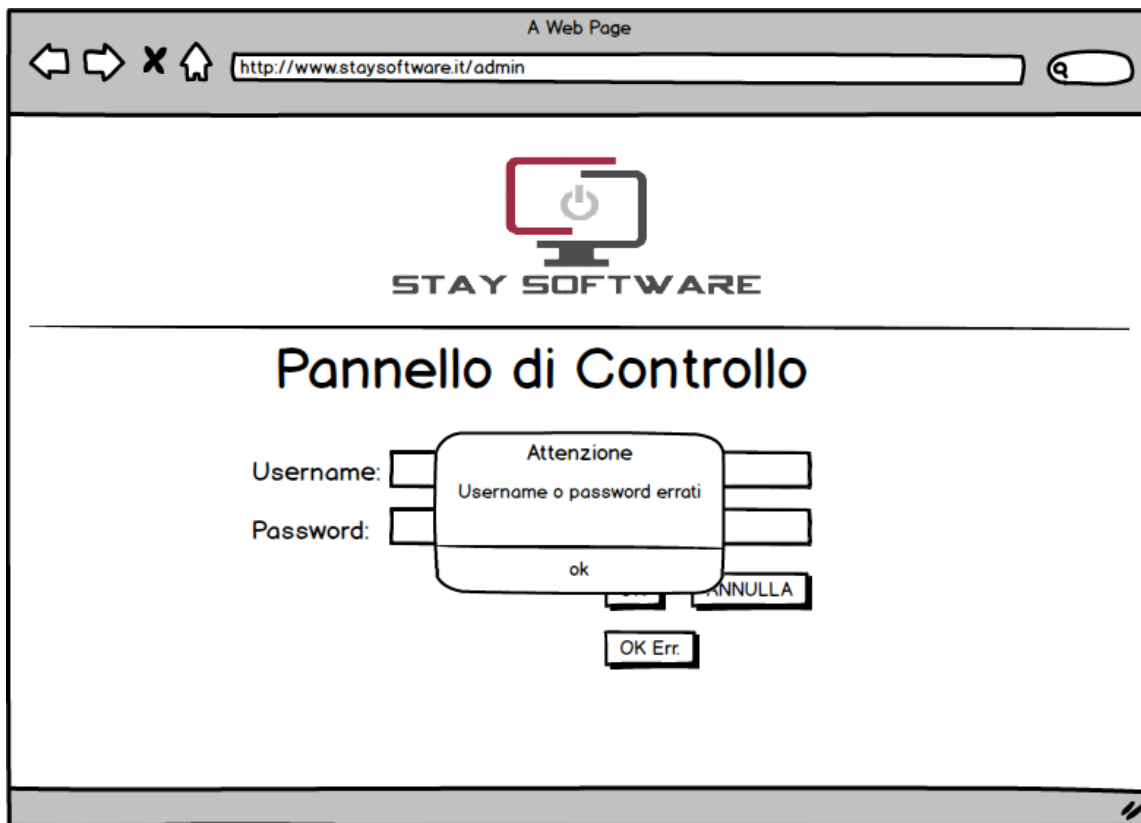
## Pannello di Controllo

Username:

Password:



## 1.1 Errore



3.



### Object Entity, Boundary, Control

Object Entity	
<b>Oggetto</b>	<b>Descrizione</b>
<i>Cliente</i>	Contiene gli attributi del cliente
<i>Amministratore</i>	Contiene gli attributi dell'amministratore
Object Boundary	
<b>Oggetti</b>	<b>Descrizione</b>
Button_accedi	Passa le informazioni inserite nel form, al controllore che effettua l'interrogazione al database
Button_registrati	Porta alla pagina di registrazione
Login_form	Form per l'inserimento dei dati
Object Control	
<b>Oggetti</b>	<b>Descrizione</b>
Login	Controlla la correttezza formale dei dati inseriti nel Login_form e interroga il database